Photon Mapping

Mayank Wadhawan, Piyush Johar, Suryansh Singh

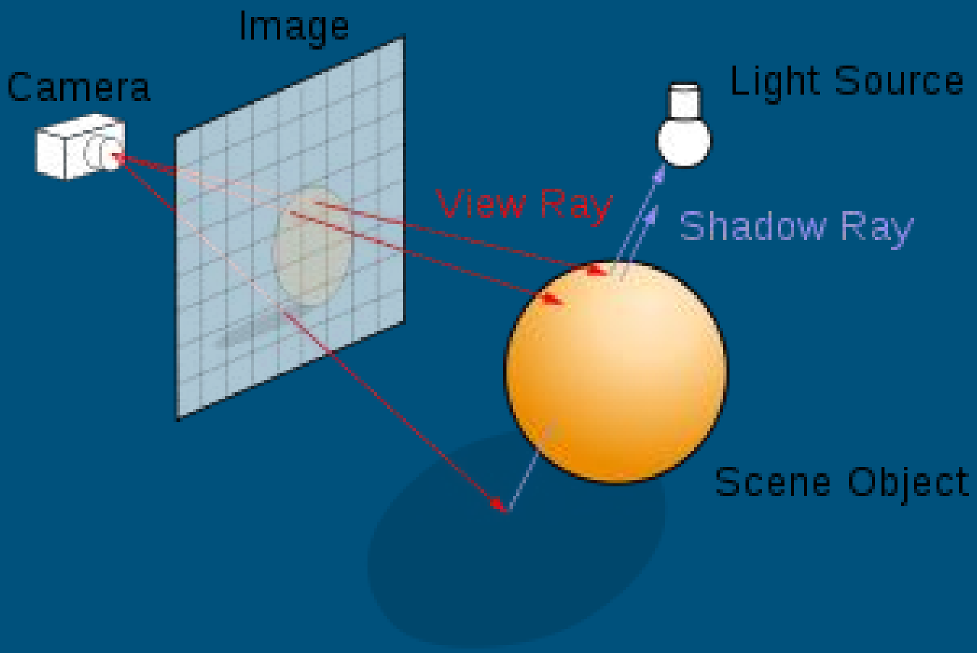
Ray Tracing

- Ray Tracing is a technique for generating an image by tracing the path of light through pixels in an image plane and simulating the effects of its encounters with virtual objects.
- The technique is capable of producing a very high degree of visual realism, usually higher than that of typical scanline rendering methods, but at a greater computational cost.

Ray Tracing

- Ray Tracing is best suited for applications where the image can be rendered slowly ahead of time, such as in still images and film, and poorly suited for real-time applications like video games where speed is critical.
- Ray tracing is capable of simulating a wide variety of optical effects, such as reflection and refraction, scattering, and dispersion phenomena

Ray Tracing



Limitation of Ray Tracing

- Given an incoming light direction and wavelength, it is possible to determine the amount of light received in any outgoing direction. This mapping is called the surface's BRDF (Bidirectional Reflectance Distribution Function). It assumes the ray will come out of the same point it reached the surface i.e. no successions of micro bounces caused it to leave the surface at another point.

Limitation of Ray Tracing

- In traditional backwards ray tracing it is not practical to simulate diffuse bounces when the ray leaves the surface in a random direction, simply because the light's rays are traced backwards from the eye to the light.
- This makes ray tracing limited in the images it can produce. Any effect that involves a diffuse bounce before the light reaches the eye is not possible in standard ray tracing.

Photon Mapping

- Photon mapping is an elegant way to solve the above limitations.
- Before any standard Ray Tracing is done, light (photons) is sent from the light sources in the scene and allowed to bounce around, then stored in a "photon map". The photon map stores final positions of the photons as well as color and incoming direction.

Photon Mapping

- After the map is built, standard Ray Tracing is done, and for each bounce the photons near the location of the bounce are essentially treated as small light sources. This allows to "complete" light paths that would not normally be considered.

Photon Mapping

- Diffuse interreflections come from light that bounces around the scene, "grabbing" color at each step.
- To simulate this, photons are stored at each diffuse bounce, but are still allowed to continue on their path, until they get absorbed.

Basic Idea

Photon Mapping tries to decouple the representation of a scene from its geometry and stores illumination information in a global data structure, the photon map.

Two-pass Algorithm

- The first pass builds the photon map by tracing photons from each light source.
- Second pass renders the scene using the information stored in the photon map.

First Pass

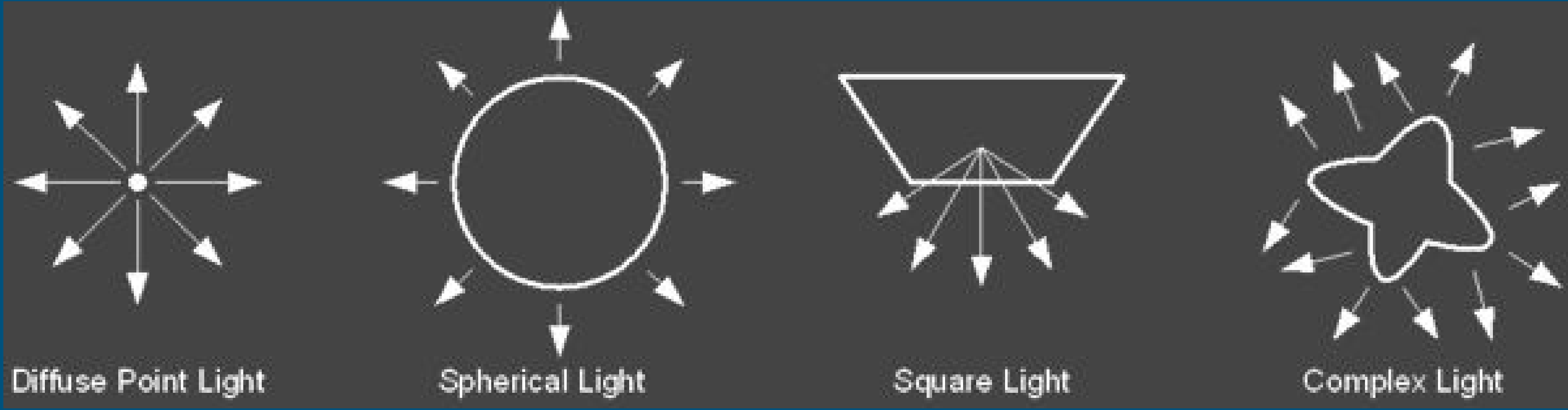
Consists of three steps:

- Photon emission.
- Photon scattering.
- Photon storing

Photon Emission

- The process of emitting discrete photons from the light sources and tracing them through the scene.
- Primary goal of this pass is to populate the photon maps that are used in the rendering pass to calculate the reflectance at surfaces and out-scattered radiance in participating media.
- When a photon hits an object, it can be reflected, transmitted, or absorbed.

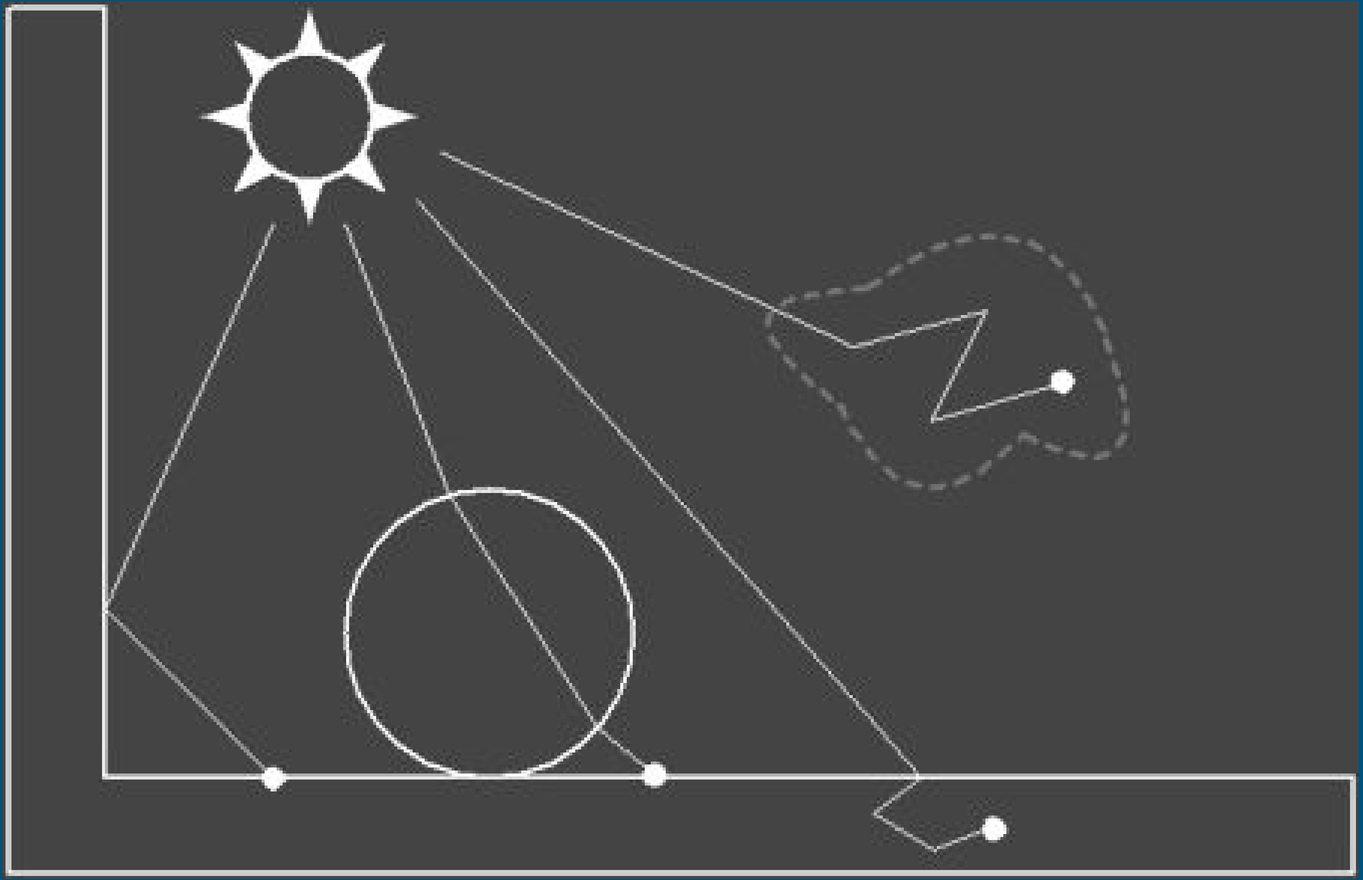
Photon Emission - Light sources



Photon Scattering

- Emitted photons from light sources are scattered through a scene and are eventually absorbed or lost.
- When a photon hits a surface we can decide how much of its energy is absorbed, reflected and refracted based on the surface's material properties.

Photon Scattering



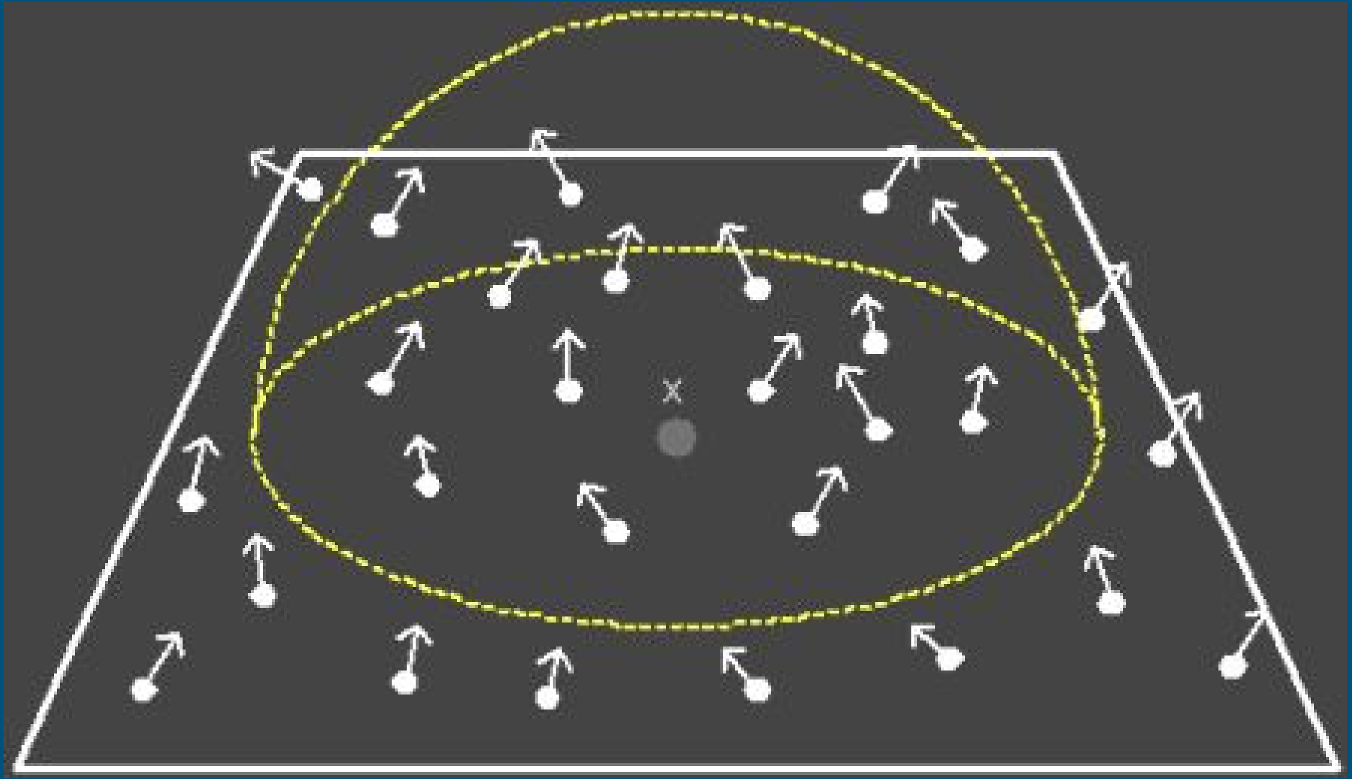
Photon Storing

- Position, color and direction of photons in the scene have to be stored.
- A good data structure is necessary as it is important to locate photons in the neighborhood of the ray-intersection.

Second Pass

- Renders the scene with the help of the photon map built in the first pass.

Rendering



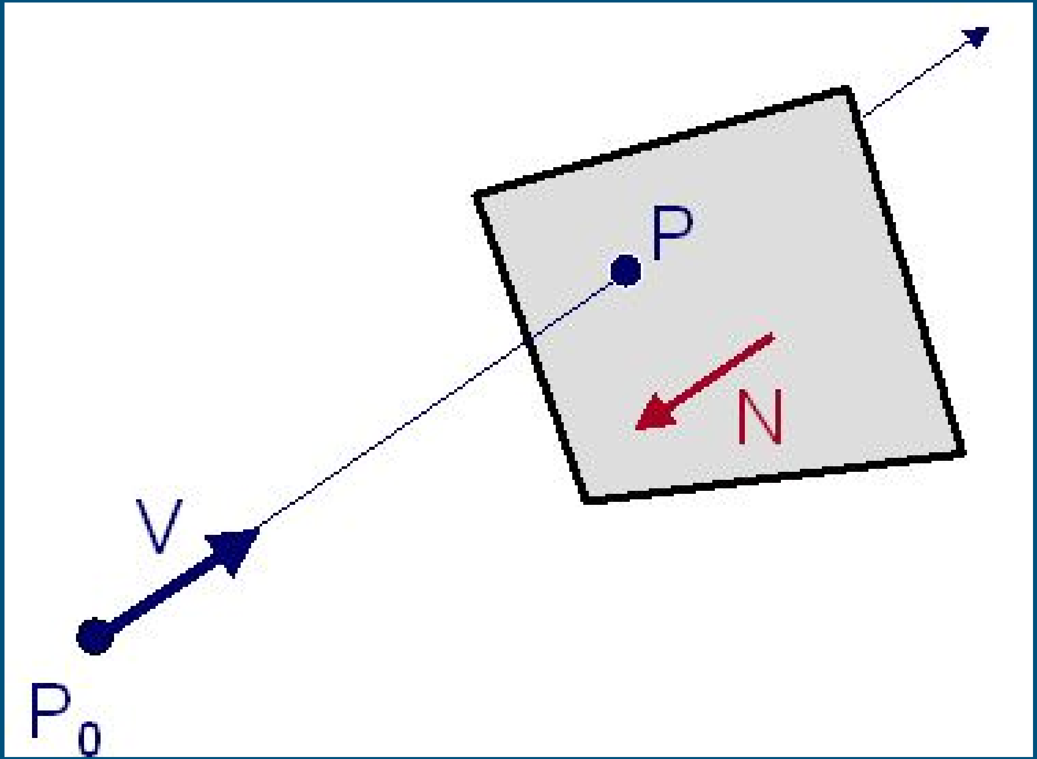
Ray Sphere Intersection

- Equation of sphere $x^2+y^2+z^2=R^2$, where x,y,z are the center and R is the radius.
- Equation of ray is p + tD where p is origin, t is the distance and D is direction.
- Solve the above 2 equations to get the point of intersections.

Ray Plane Intersection

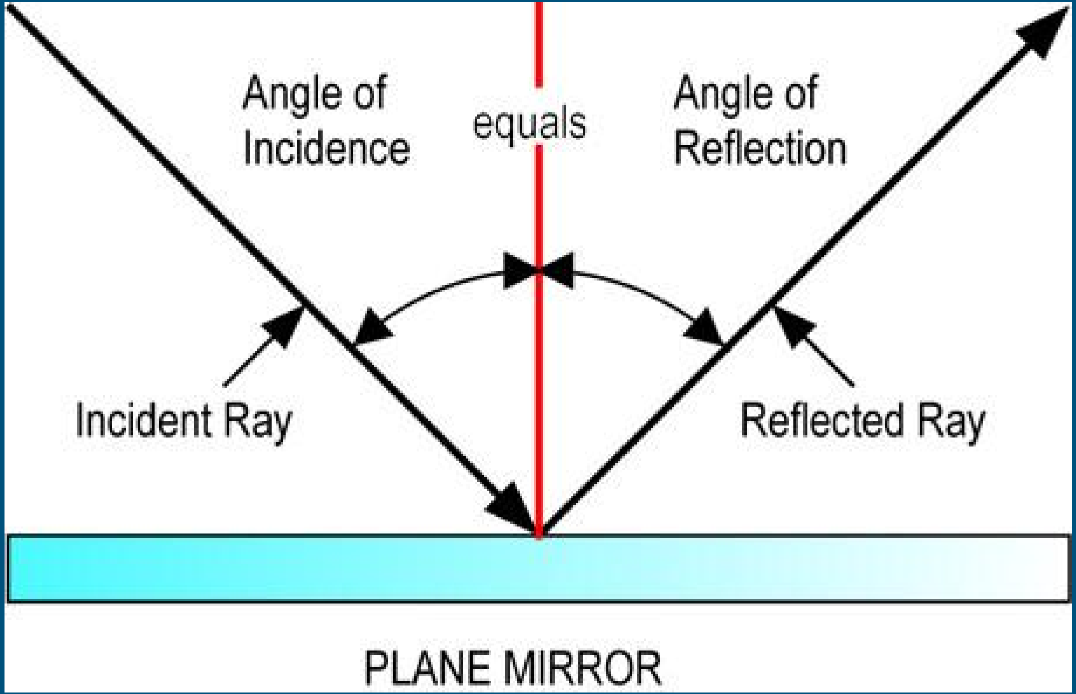
- Equation of ray is p + tD where p is origin, t is the distance and D is direction.
- Equation of plane is P.N + d.
- Substituting equation of ray in plane, we solve
- (p+tD).N + d to get the point of intersection.

Ray Plane Intersection



Reflection

$$\begin{aligned}\mathbf{r} &= \mathbf{i}_{\|} - \mathbf{i}_{\perp} \\ &= [\mathbf{i} - (\mathbf{i}\cdot\mathbf{n})\mathbf{n}] - (\mathbf{i}\cdot\mathbf{n})\mathbf{n} \\ &= \mathbf{i} - 2(\mathbf{i}\cdot\mathbf{n})\mathbf{n}\end{aligned}$$



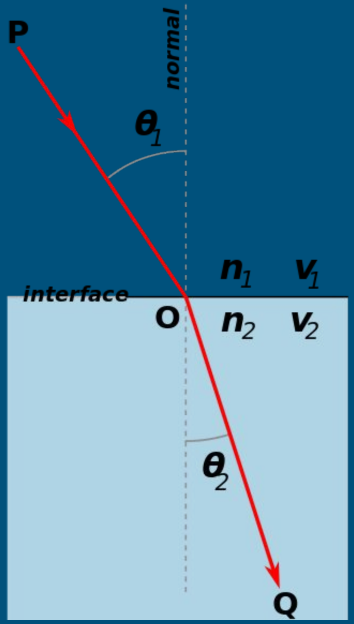
Refraction

Snell’s law:

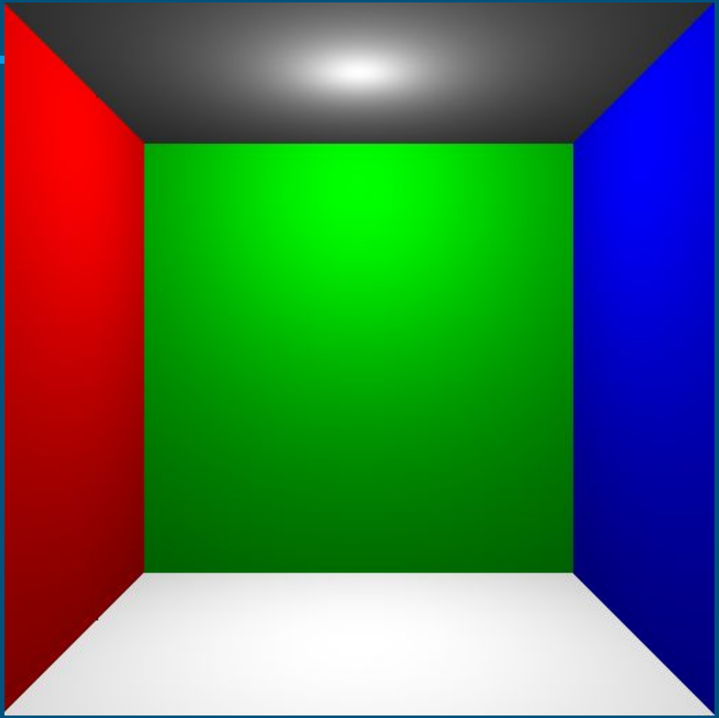
Snell's law is used to describe the relationship between the angles of incidence and refraction, when referring to light or other waves passing through a boundary between two different isotropic media, such as water, glass, or air.

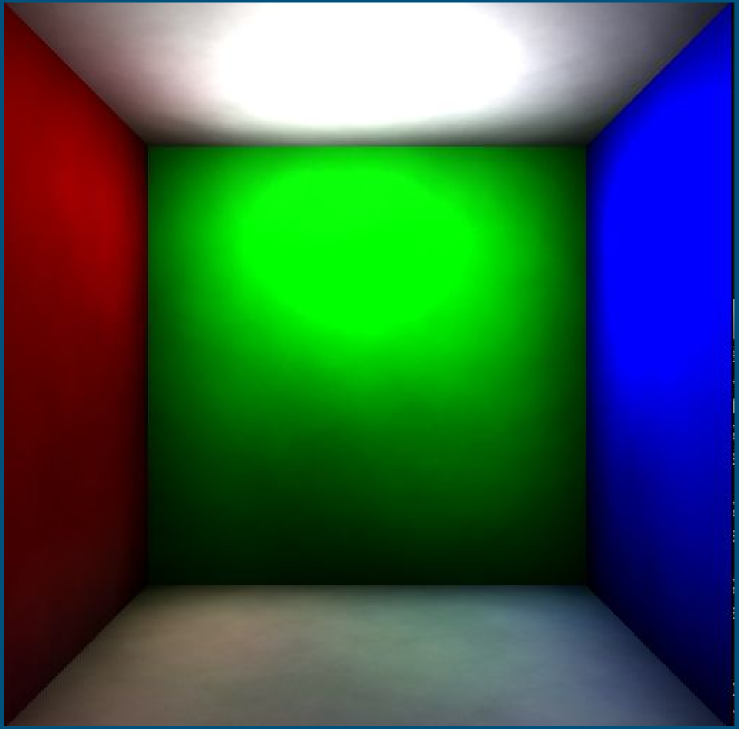
Refraction

$$\mathbf{v}_{\text{refract}} = r\mathbf{l} + \left(rc - \sqrt{1 - r^2\left(1 - c^2\right)}\right)\mathbf{n}$$

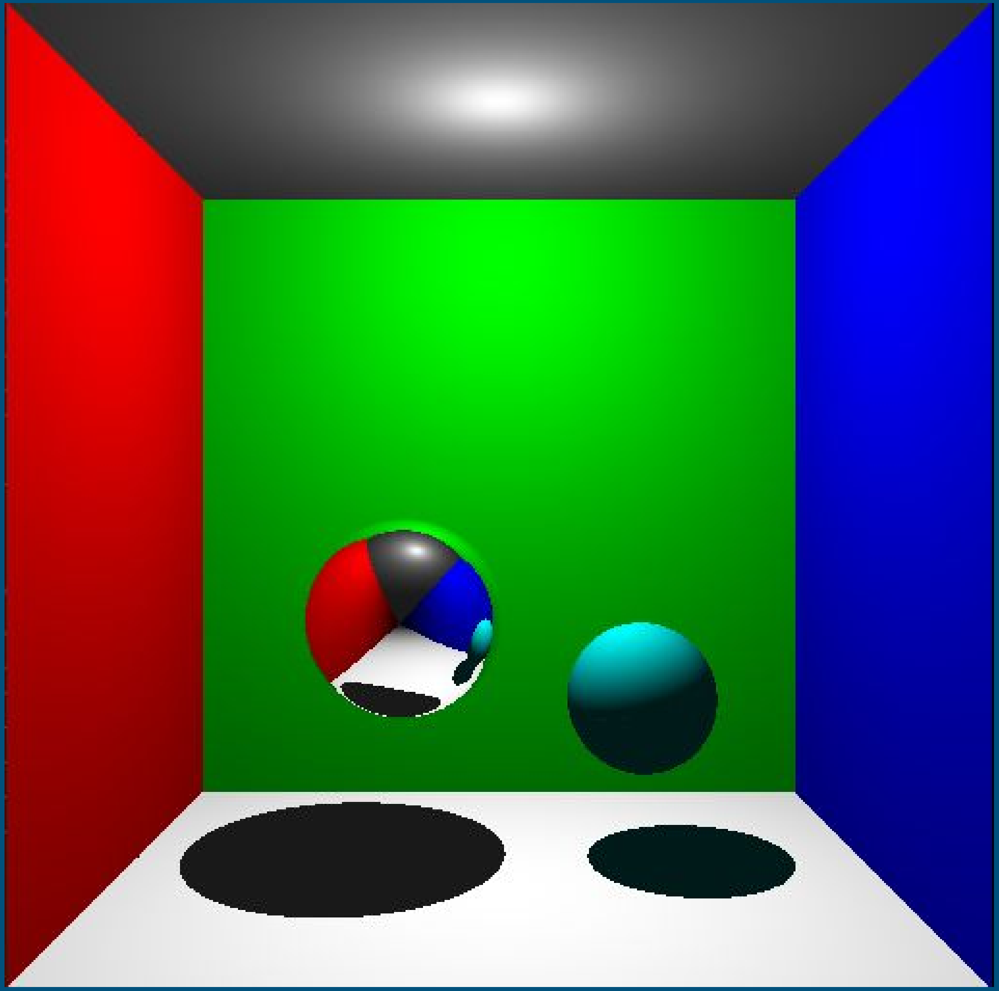


Environment Setup - Cornell Box

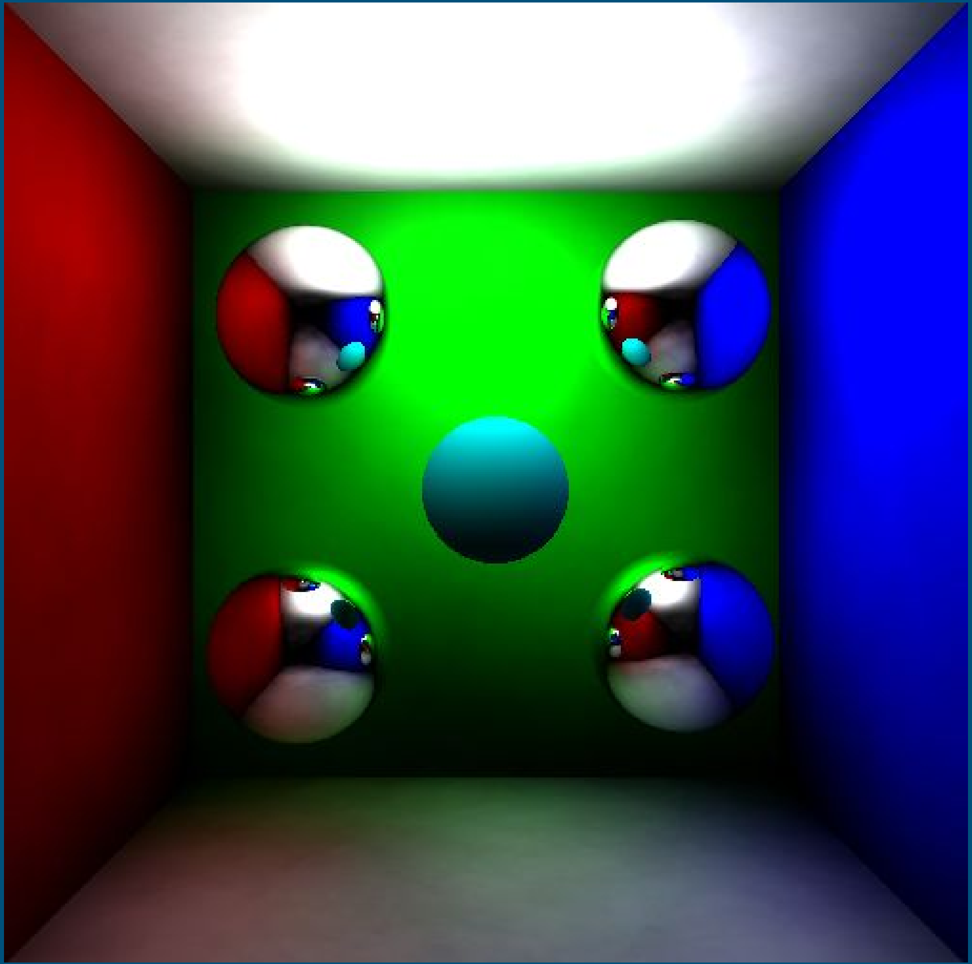

Ray Tracing

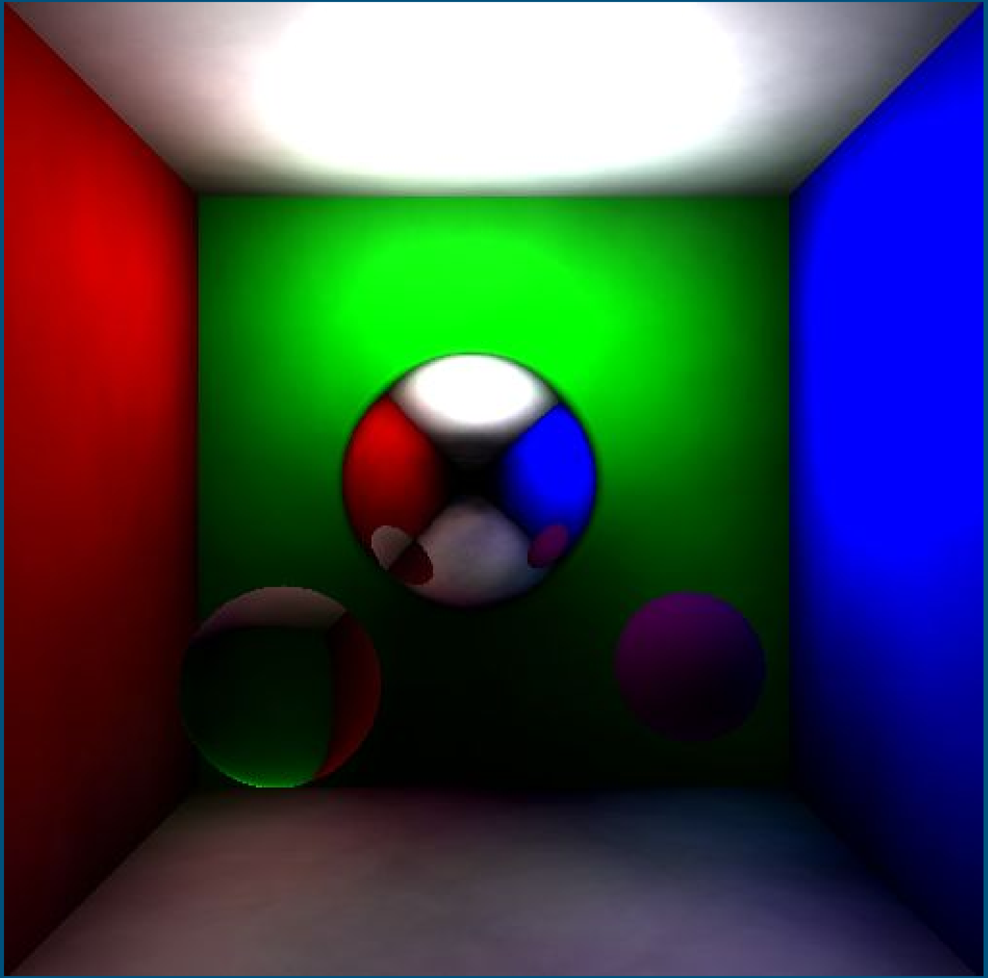

Photon Mapping

Ray Tracing

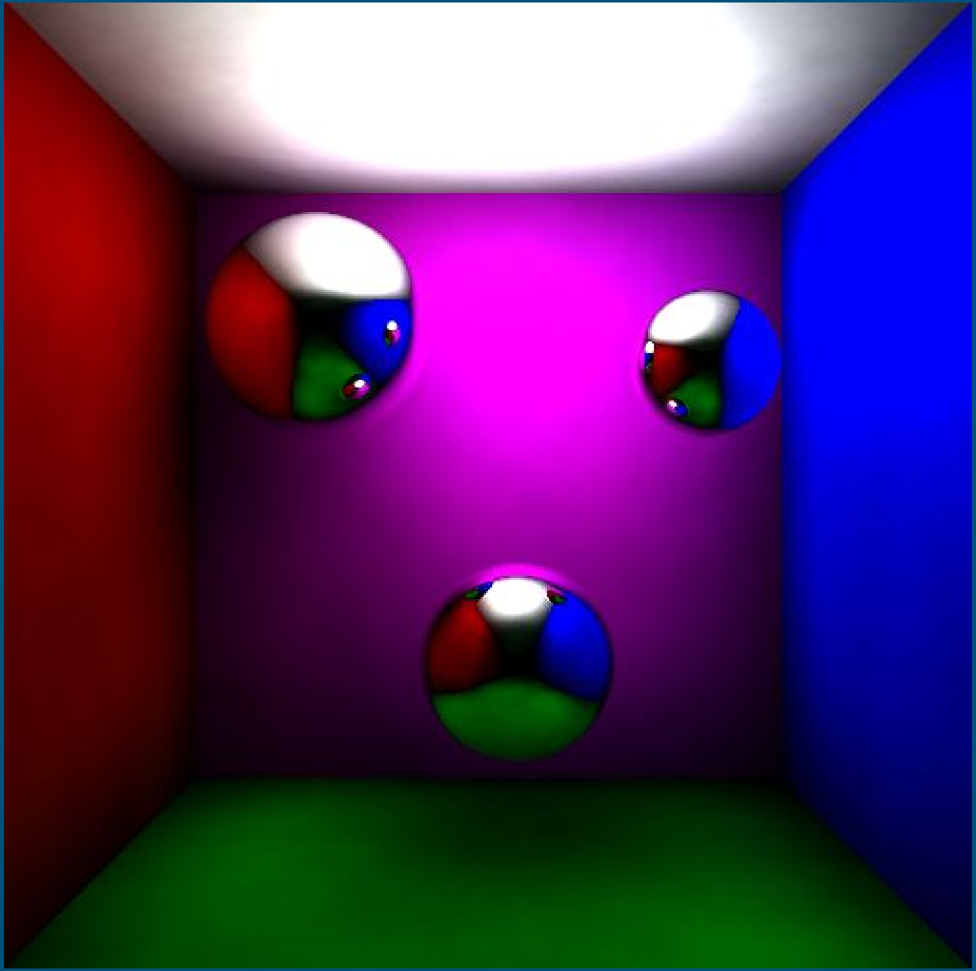


Photon Mapping

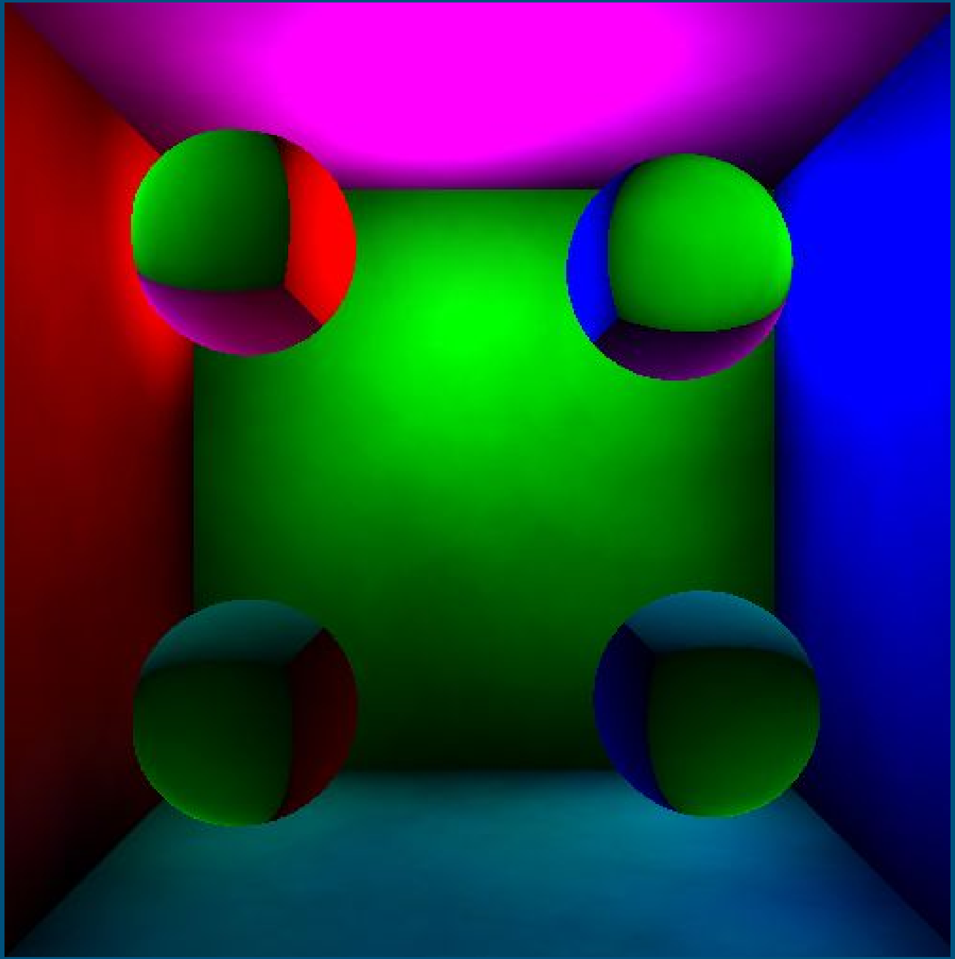




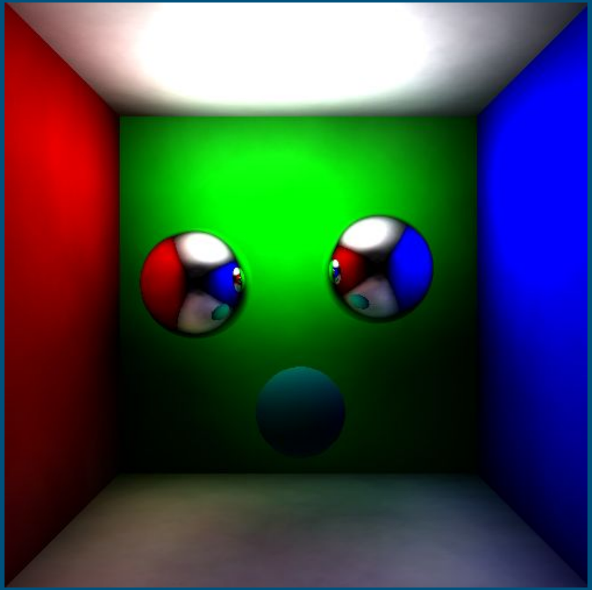
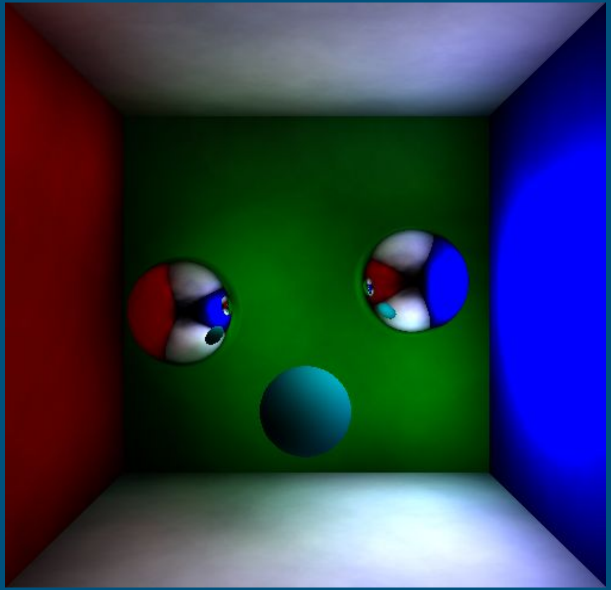
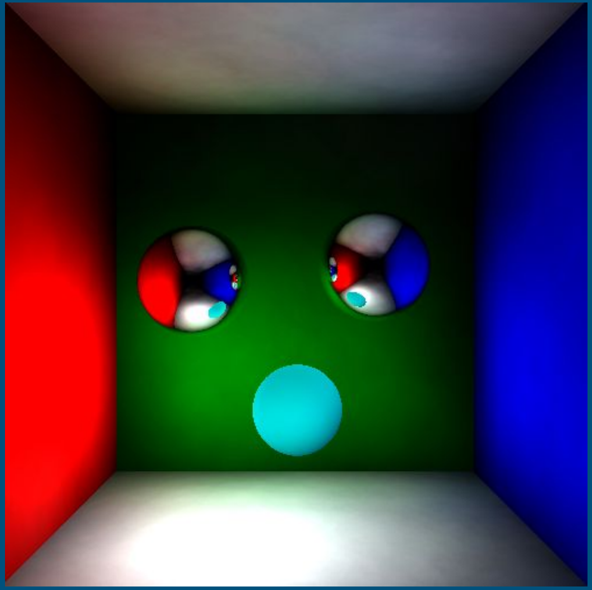
Reflection



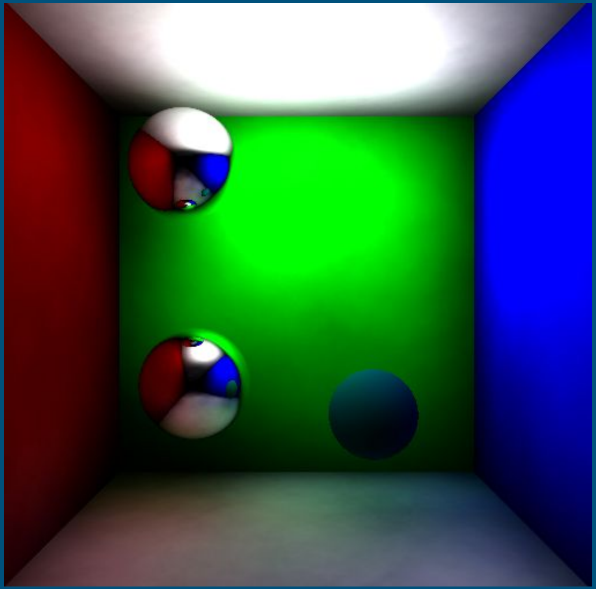
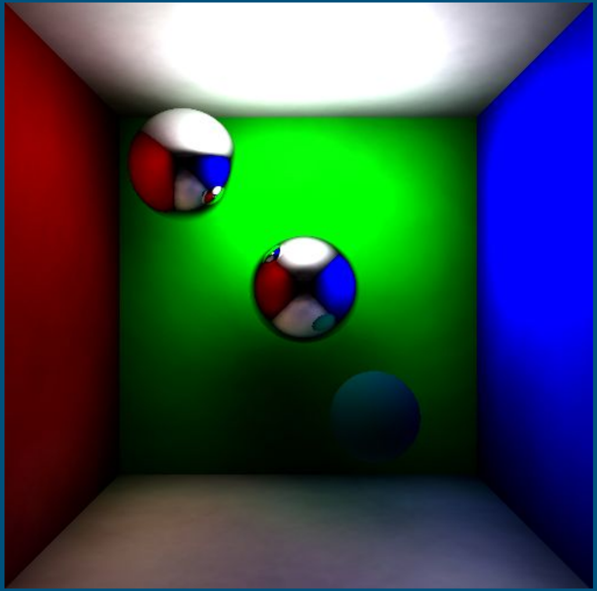
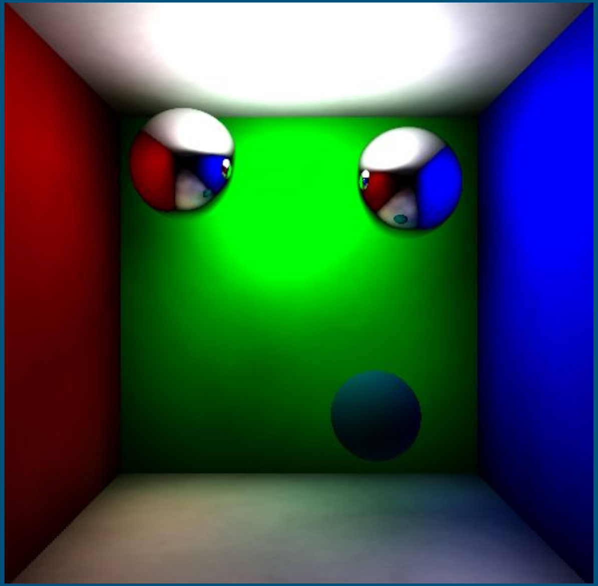
Refraction



Moving the light source

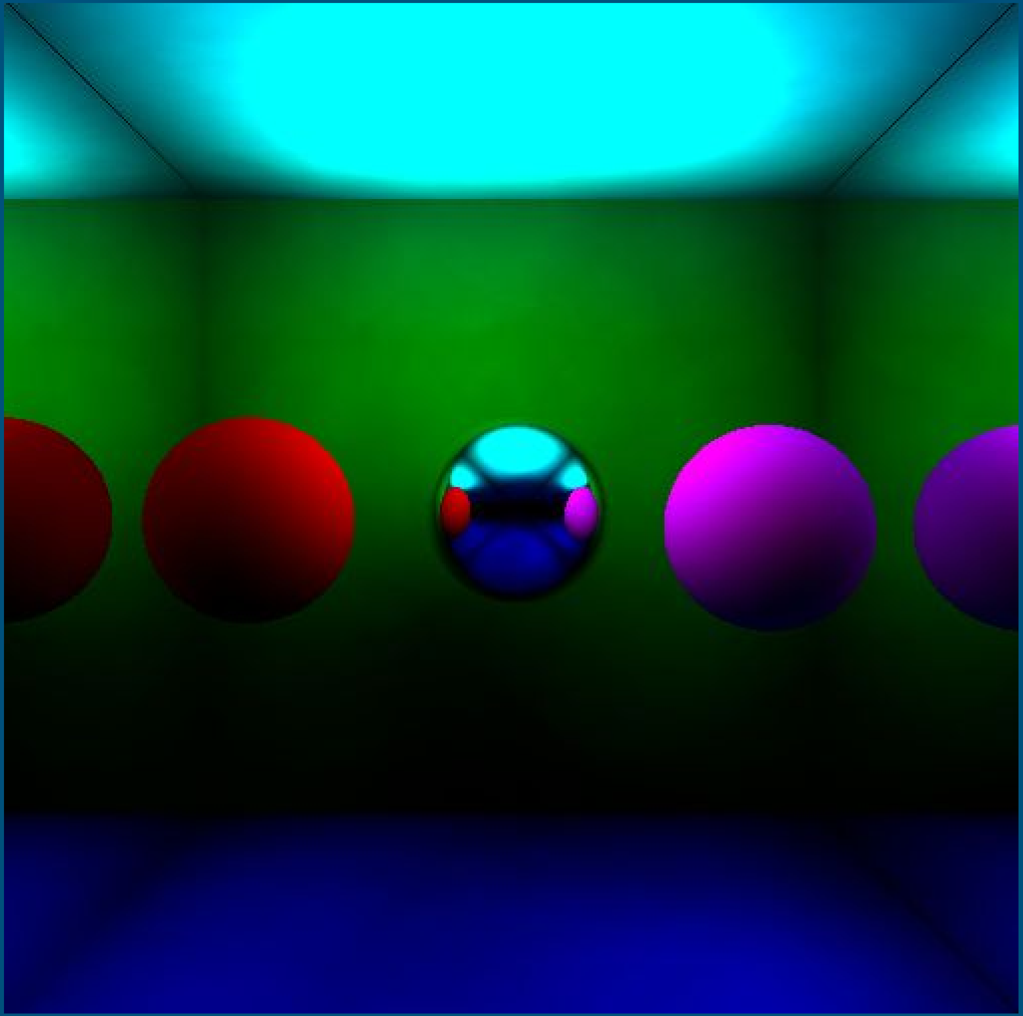




Moving of Objects

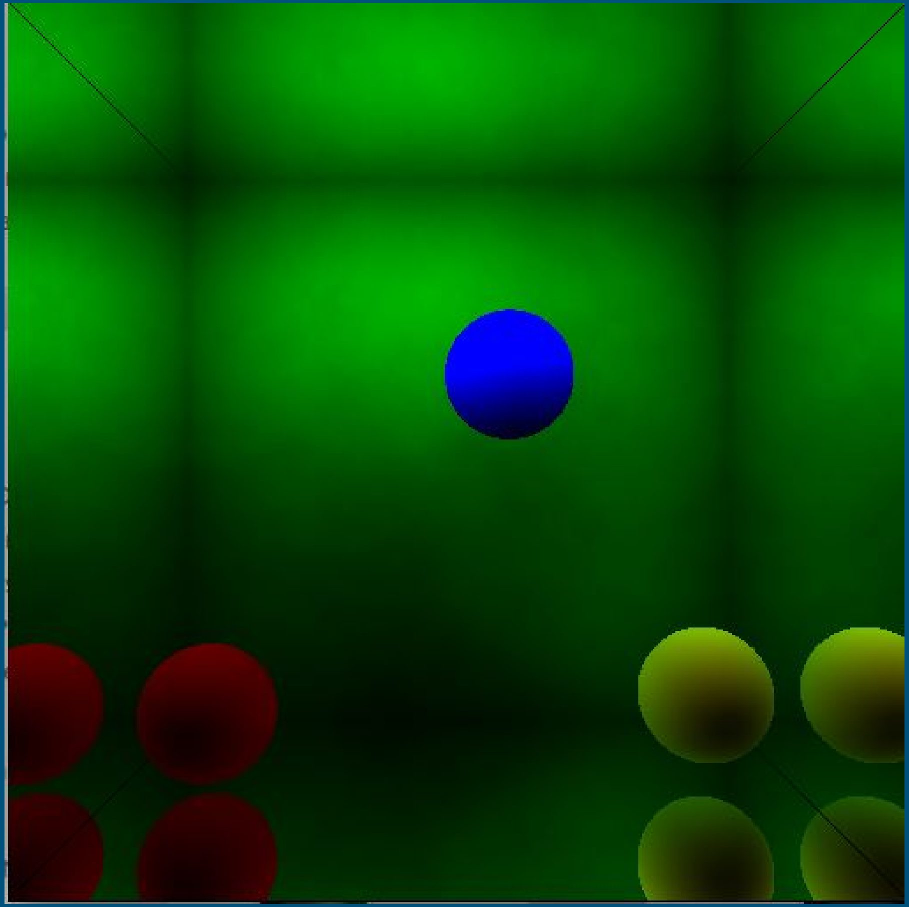
Additional Work

Creating a mirror effect



Additional Work

Creating a mirror effect



References

http://graphics.ucsd.edu/~henrik/papers/photon_map/global_illumination_using_photon_maps_egwr96.pdf

http://marctenbosch.com/photon/

https://graphics.stanford.edu/courses/cs348b-01/course8.pdf

Acknowledgment

Dr. Toler-Franklin for her guidance and assistance.

Thank You